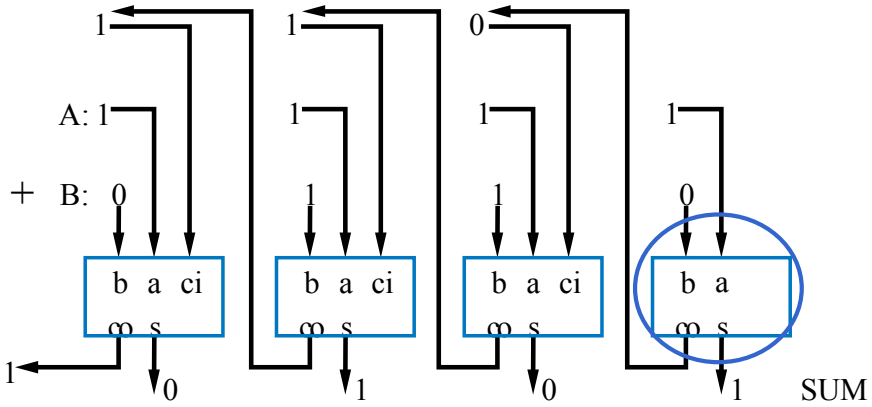
Adders

- Adds two N-bit binary numbers
  - 2-bit adder: adds two 2-bit numbers, outputs 3-bit result
  - e.g., 01 + 11 = 100 (1 + 3 = 4)
- Can design using combinational design process of Ch 2, but doesn't work well for typical N
  - Why not?

Inputs				Outputs		
a1	a0	b1	b0	c	s1	s0
0	0	0	0	0	0	0
0	0	0	1	0	0	1
0	0	1	0	0	1	0
0	0	1	1	0	1	1
0	1	0	0	0	0	1
0	1	0	1	0	1	0
0	1	1	0	0	1	1
0	1	1	1	1	0	0
1	0	0	0	0	1	0
1	0	0	1	0	1	1
1	0	1	0	1	0	0
1	0	1	1	1	0	1
1	1	0	0	0	1	1
1	1	0	1	1	0	0
1	1	1	0	1	0	1
1	1	1	1	1	1	0

Half-Adder

- *Half-adder*: Adds 2 bits, generates sum and carry
- Design using combinational design process from Ch 2



Step 1: Capture the function

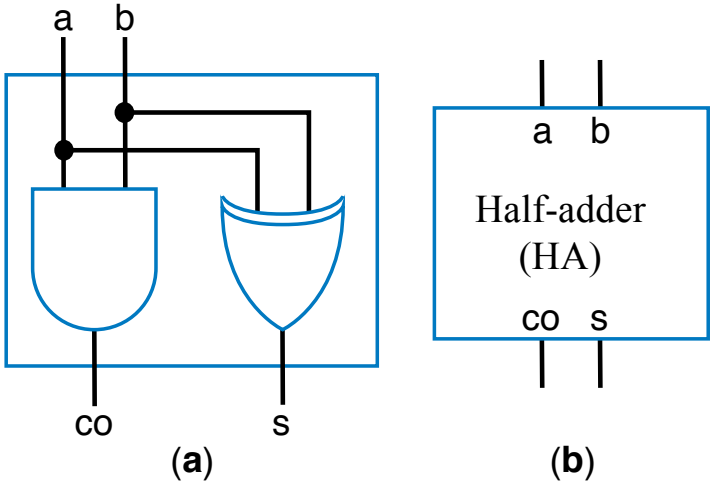
Inputs		Outputs	
a	b	co	s
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Step 2A: Create equations

co = ab

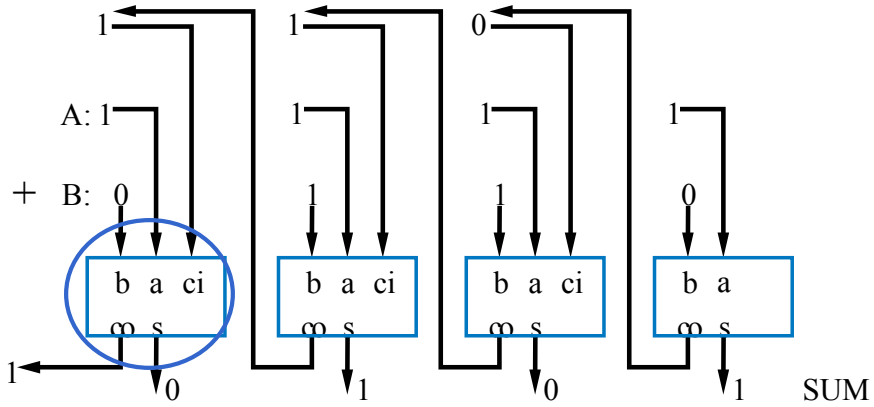
s = a'b + ab' (same as s = a xor b)

Step 2B: Implement as circuit



Full-Adder

- *Full-adder:* Adds 3 bits, generates sum and carry
- Design using combinational design process from Ch 2



Step 1: Capture the function

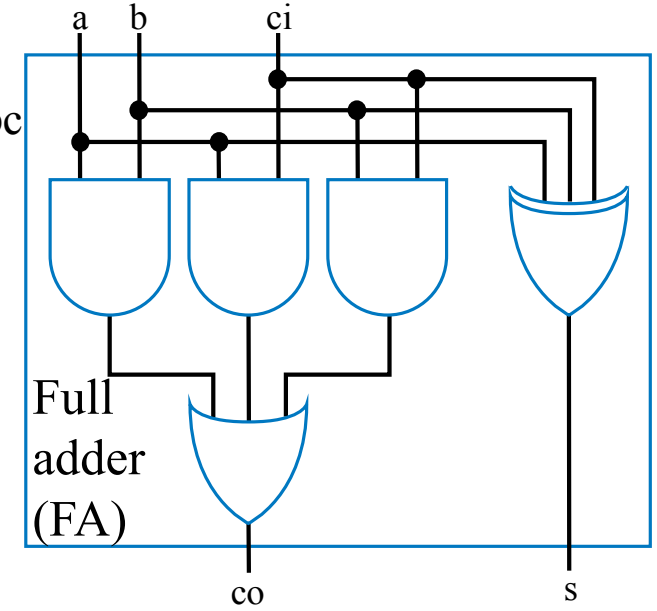
Inputs			Outputs	
a	b	ci	co	s
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Step 2A: Create equations

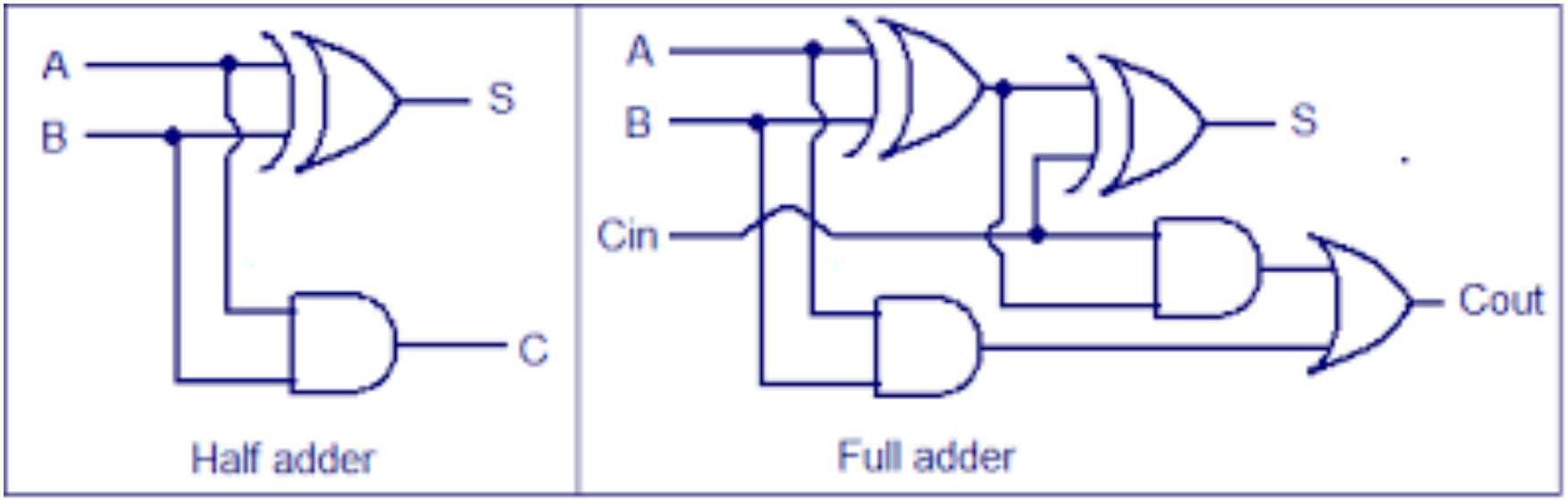
co = a'bc + ab'c + abc' + abc
co = a'bc +abc +ab'c +abc +abc' +abc
co = (a'+a)bc + (b'+b)ac + (c'+c)ab
co = bc + ac + ab

s = a'b'c + a'bc' + ab'c' + abc
s = a'(b'c + bc') + a(b'c' + bc)
s = a'(b xor c)' + a(b xor c)
s = a xor b xor c

Step 2B: Implement as circuit



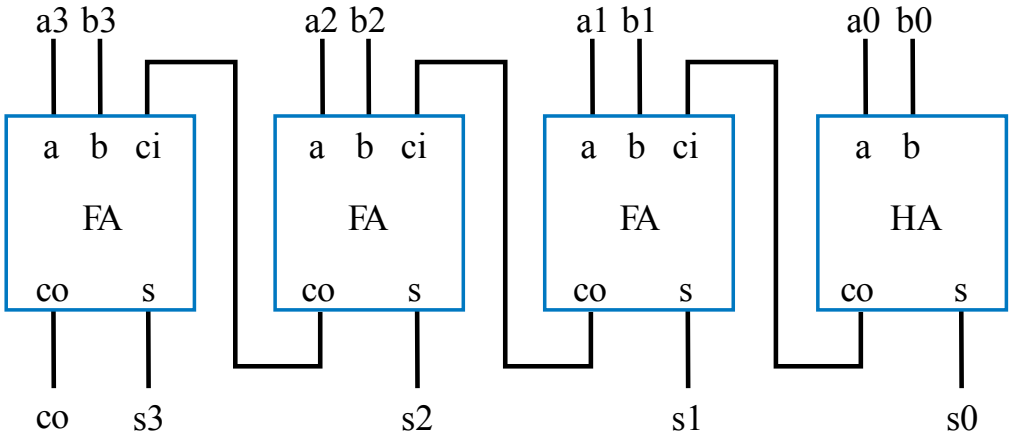
Full-Adder using Half Adder



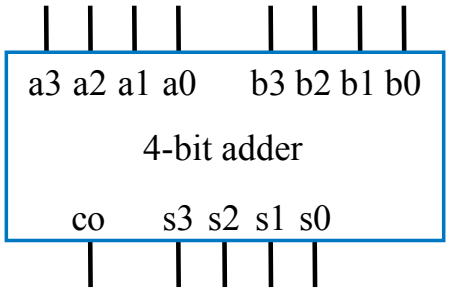
Carry-Ripple Adder

- Using half-adder and full-adders, we can build adder that adds like we would by hand

- Called a *carry-ripple adder*
  - 4-bit adder shown: Adds two 4-bit numbers, generates 5-bit output
    - 5-bit output can be considered 4-bit "sum" plus 1-bit "carry out"
  - Can easily build any size adder



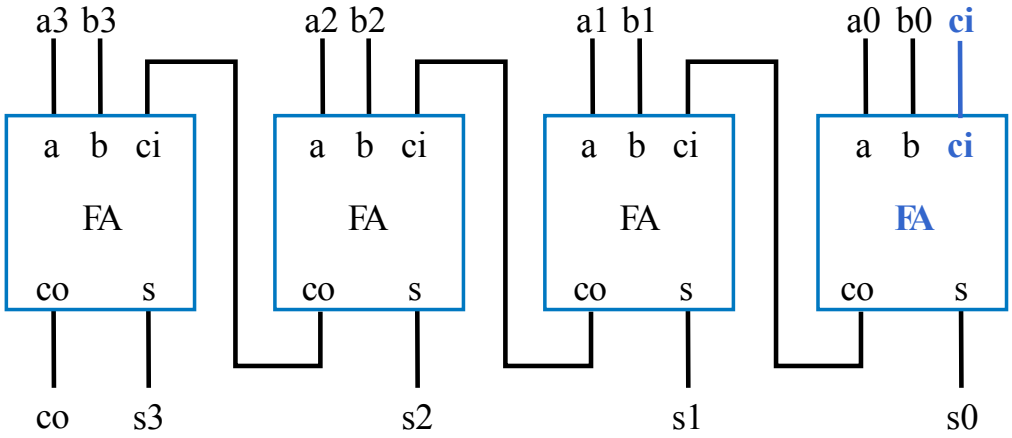
(a)



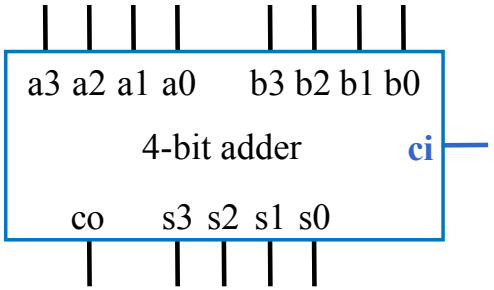
(b)

Carry-Ripple Adder

- Using full-adder instead of half-adder for first bit, we can include a "carry in" bit in the addition
  - Useful later when we connect smaller adders to form bigger adders



(a)



(b)